6-2019

Exploring the Behavior Repertoire of a Wireless Vibrationally Actuated Tensegrity Robot

Zongliang Ji
Union College - Schenectady, NY

Exploring the Behavior Repertoire of a Wireless Vibrationally Actuated Tensegrity Robot

By

Zongliang (Jerry) Ji

* * * * * * * * *

Submitted in partial fulfillment

of the requirements for

Honors in the Department of Computer Science

UNION COLLEGE

March, 2019

Abstract

ZONGLIANG (JERRY) JI Exploring the Behavior Repertoire of a Wireless Vibrationally Actuated Tensegrity Robot. Department of Computer Science, March, 2019.

ADVISOR: John Rieffel

Soft robotics is an emerging field of research due to its potential to explore and operate in unstructured, rugged, and dynamic environments. However, the properties that make soft robots compelling also make them difficult to robustly control. Here at Union, we developed the world's first wireless soft tensegrity robot. The goal of my thesis is to explore effective and efficient methods to explore the diverse behavior our tensegrity robot. We will achieve that by applying state-of-art machine learning technique and a novelty search algorithm.

Contents

List of Figures

List of Tables

1 Introduction

Imagine a robot on a rescue mission right after a disastrous earthquake. The robot carries both injured people and precious cargo worth millions of dollars. Suddenly, a small wave of aftershocks covers the robot with rocks and dust. However, since the robot is resilient to shape changes, it protects the people and the cargo inside and continues moving towards its designated location.

The robot we are describing is not a traditional type of robot with limbs, arms, or wheels. It is a type of soft robot called a “tensegrity”, that is inspired by natural biology. “Tensegrity” is a word that combines “tensile” and “integrity”. It was first coined in the late 1900s by Buckminster Fuller to describe sculptures created by the sculptor Kenneth Snelson [11]. Since then, tensegrity structures became famous in art, architecture, and science. A tensegrity structure is a self-supporting structure consisting of a set of disjoint rigid elements (known as rods or struts) whose endpoints are connected by a set of continuous tensile elements (known as strings, cables, or springs).

Due to the balance between the tensile and compression forces in the structure, a tensegrity structure is able to maintain its shape by reaching an equilibrium state [41]. The compressive and tensile elements in the structure need not resist significant bending or shear forces. This means that a tensegrity structure has a tendency to return to its original stable configuration after any moderate and temporary external forces are applied [8]. Because of their useful properties, tensegrity structures have been widely adopted in civil engineering as the basis of extremely light weight, yet strong mechanical structures using little material. In architecture, tensegrity structures are used in bridges and geodesic domes [12].

The property of handling external forces applied from different directions also suggests that tensegrity structures are well suited to operating in dynamic environments where contact forces are not easy to predict. Frequently, researchers find tensegrity structures in biology, such as the muscular-skeletal system [22], the animal cell’s cytoskeleton [13], and the mechanical system of human body [25]. The field of soft robotics has brought tensegrity structures into its genus of biologically-inspired robot morphologies.

NASA is one of the few organizations that is invested in tensegrity research. The main reason for NASA’s interest is the potential of tensegrity robots in space exploration programs; NASA hopes to use tensegrity robotics as planetary rovers. The high strength-to-weight ratio of tensegrity robots is attractive because it reduces launch mission expenses. Due to the high resilience and natural form-finding properties of tensegrity structures, large tensegrity robots are deployable from compact configurations, enabling them to fit into space-constrained spacecraft. While these qualities have inspired studies of deployable antennae, other large tensegrity robots have only recently been investigated for planetary exploration [39].

One key goal of NASA’s work is to develop a tensegrity probe with an actively controlled tensile network, enabling compact stowage for launch, followed by deployment for landing. Tensegrity probes can safely absorb significant impact forces, enabling high-speed entry, descent and landing scenarios, where the probe acts like an airbag. However, unlike rovers that are deployed inside airbags that are discarded after a single use, tensegrity robots can be launched without airbags and also provide rolling mobility. This enables compact and lightweight planetary exploration missions with the capabilities of traditional wheeled rovers, but with mass and cost similar to a stationary probe. The dual use of the structure reduces mission cost and enables new forms of surface exploration using the tensegrity's natural impact tolerance [39].

However, these excellent qualities of tensegrities also carry complex nonlinear dynamics with them. Actively controlling a tensegrity robot is a difficult task yet effective locomotion is the most crucial element of robotics. In this research, I will explore the diverse behavior of a wireless vibrationally actuated tensegrity robot that is designed and built at Union College to reach the goal of controlling the robot.

In Union College's Evolutionary Robotics lab, we have been developing our own tensegrity robots under the guidance of Professor John Rieffel. Previous iterations of our tensegrity robots achieved effective locomotion by vibrating rigid struts on the robot. However, the old design of our tensegrity robot used magnetic wires connected to each strut which made it hard to conduct experiments outside the lab area. For the last two years, we have been working on designing and implementing a new wireless iteration of our vibrationally actuated limbless tensegrity robot known as VVValtr. With on-board circuitry and wireless RFduino chip communication, we now have a new wireless tensegrity robot in our lab. My hope is to explore the new wireless robot and enable the robot to achieve effective locomotion, to conduct more interesting research with a fully functional tensegrity robot. Eventually, we want to enable the tensegrity robot to conduct rescue missions. The goal of my research is to effectively explore the behavior of tensegrity robot by combining rotations and translations. I will also compare different solutions of moving our tensegrity and analyze their performance using metrics like time efficiency and energy consumption.

2 Background

A tensegrity structure is a mechanical structure based on a subtle interplay between compressive and tensile forces [36]. A general definition by Pugh et al. is that: *A tensegrity structure is a structure that maintains a stable volume in space through the use of discontinuous compressive elements (struts) connected to a continuous network of tensile elements (cables) [27].*

Recently, apart from conventional design methodology, comprised of a series of rigid links connected by prismatic joints, researchers have started to explore a new paradigm in the mechanical design of robots, based on tensegrity structures [26]. They define this new type of robot as: *Tensegrity robots are interlocked* *sets of disjoint rigid struts and tensional connections such as springs or cables [26].*

As there are no lever arms like classical hard robots, forces are not applied directly on joints or other common areas. Instead, external forces are distributed through out the structure via multiple load paths. This helps tensegrity robots create mechanical robustness and tolerance to forces applied from any direction.

There are various designs of tensegrity robots, each with their own control mechanisms. Normally, the control of a tensegrity robot requires a control architecture and an optimization algorithm to find the optimal way to move the tensegrity robot effectively. Researchers have conducted experiments on tensegrity robots in virtual simulations as well as in the physical world. Based on the different designs of the robots, the type of control can either be a closed-loop or open-loop control.

Almost all work in locomotion research and tensegrity robotics designs starts with simulations. The two frequently used simulation tools are Open Dynamic Engine (ODE) library used by Paul et al. [26] and Rieffel et al. [33] and the Bullet Physics engine used by NASA [5] [39].

After achieving promising results in simulation, researchers then design and implement the physical tensegrity to test their control architecture and optimization algorithms.

Following the work done by Paul et al. [26], there are three ways tensegrity robots move:

- Crawling is the earliest way of moving proposed by Paul et al [26]. In Rieffel et al.'s morphological communication [33], the robots also move by crawling in the simulated world. Hirai et al. also researched tensegrities that crawl [35].

- Rolling locomotion is mostly supported by the NASA research team [4] [34]. It was also studied by Hirai et al. in 2012 [21].

- The vibrational moving strategy was a novel idea proposed by Rieffel et al. [17] and Bohm et al. [3]. The idea is to vibrate the rigid bars of the tensegrity robots to achieve locomotion.

There are three main categories of controllers used by researchers.

- The first category is based mostly on the design of the tensegrity robots. For instance, Paul et al. used servomotors as the controller of the robot by compressing the length of the cables [26]. Rieffel et al. used motors installed on the rigid struts to vibrate the struts given the motor frequency commands [18].

- The second is the Spiking Neural Network architecture that Rieffel et al. presented in the morphological communication work [33].

- The third is the Central Pattern Generator (CPG) proposed by Bliss et al. [1] which is currently implemented by NASA [4].

One of the crucial features for achieving locomotion in robotics is the type of loop the robot uses to move. There are two types of control:

1. Open-loop control means that the robot is not sensing the environment while it is moving. For example, this is like driving a car at a constant speed without looking at the speedometer.

2. Closed-loop control means that the robot can change its behavior using feedback from environmental sensors. This is like driving a car by checking for other cars on the road, traffic lights, and the road surface, and then adjusting the speed of the car accordingly.

The simpler the design of the robot and the fewer the sensors, the more likely the control strategy is open-loop. For instance, the tensegrity robots implemented by Paul et al. use an open-loop design to achieve locomotion [26].

Closed-loop designs are mostly implemented and tested in the simulated environments similar to the morphological communication architecture [33] and CPG architecture [1]. So far, only two groups have tested their physical robots using closed-loop control. Bliss et al. [1] tested the closed-loop CPG on a more restricted physical design of a tensegrity robot. NASA's team tested CPG and hybrid CPG methods on both of their ReCTeR and the SUPERball tensegrity robots [5] [34].

In addition, there are several categories of optimization algorithms used to gain effective and optimal gait of various tensegrity robots. The first category, most commonly used, utilizes different variations of evolutionary algorithms [1] [4] [26] [33]. The second category uses a various of machine learning algorithms such as Bayesian Optimization proposed by Rieffel et al. [31] and deep reinforcement learning by Zhang et al [43].

2.1 Related Work

The dynamic properties of tensegrity robots have been relatively unexplored until recent years. Early examples of kinematic motion include the work at EPFL's IMAC laboratory [10]. Skelton and Sultan introduced algorithms for the positioning of tensegrity-based telescopes and the dynamic control of a tensegrity flight simulator platform [38]. Although there were some early efforts at MIT's CSAIL lab, it wasn't until the work of Paul et al. at Cornell University made the concept of tensegrity robotics became widespread [26]. In their paper, they introduced robots based on tensegrity structures, which demonstrated that the dynamic of such structures can be utilized for locomotion [26]. Paul et al. designed two tensegrity robots, TR3, a three-strut tensegrity and TR4, a four-strut tensegrity. The experiment was conducted by running evolutionary optimizations to obtain controllers for both of the robots in an ODE simulation environment. Both robots received effective locomotion in the simulation environment. This is the first time tensegrity structures were used in developing robots, and gained effective locomotion. Here, the researchers used a genetic algorithm to find the most effective gait for the robot. Their study demonstrated that tensegrity structures can provide the basis for lightweight, strong, and fault-tolerant robots, with a potential for a variety of locomotor gaits.

In 2009, Rieffel et al. published a paper that uses morphological communication inspired by biology to design the tensegrity robot controller [33]. In this paper, the architecture that was introduced used time-sensitive spiking neural networks to enable "morphological communication." This introduces a new paradigm for decentralized control of large, coupled, modular systems. Rieffel et al.'s study involved methods of controlling both large tensegrity structures [6] and minimal structures [26]. The results of exploring different gaits in simulation demonstrated how coupled dynamic properties of tensegrity structures can be exploited to benefit movement. This study first brought up the use of morphology of robots to perform computation and communication.

Other work published in 2012 by Bliss et al [1] proposed an architecture for tensegrity robot control called Central Pattern Generator (CPG). Central pattern generators are neuronal circuits that, when activated, can produce rhythmic motor patterns such as walking, breathing, flying, and swimming in the absence of sensory or descending inputs that carry specific timing information. Its use is widely observed in the animal world. In their paper, they conducted experiments to validate synthesized CPG control of tensegrity structures. The reason CPGs are used is that the pre-stressed cables in the tensegrity structure provide a method of simultaneous actuation and sensing, which is similar to the biological motor control mechanism of regulating muscle stiffness through motoneuron activation and sensing.

From 2013, NASA has shown interest in tensegrity structures and tensegrity robots suggested by the prior work done by Skleton et al. [37] and Paul et al. [26]. Their goal is to utilize static and dynamic properties of tensegrities to land on planets like rovers in an airbag and also traverse on planets with similar capabilities to traditional wheeled rovers. Researchers from NASA first explored the learning methods for a tensegrity to move in a simulation environment. The work published by Iscen et al. in 2013 uses multiagent learning to learn how to control a ball-shaped tensegrity with 6 rods and 24 cables [14]. Their simulation results show that multiagent learning can be used to learn an efficient rolling behavior. In 2014, researchers built the first physical tensegrity robot for NASA and tested various control and optimization algorithms on it [5]. In their work, they developed a new open source simulation environment, the NASA Tensegrity Robotics Toolkit (NTRT) based on the discrete time Bullet Physics engine. They also designed and implemented a hardware prototype of a spherical six-bar tensegrity, the Reservoir Compliant Tensegrity Robot (ReCTeR).

SUPERball is a second generation prototype, evolving from the tensegrity robot ReCTeR, which is also a modular, lightweight, highly compliant 6-strut tensegrity robot that NASA used to validate their NTRT simulator. In 2015, the first phase of building and testing SUPERball was finished [34]. SUPERball is the first full prototype of this tensegrity robot platform, and is eventually destined for space exploration missions. Currently, SUPERball is fully built and under experiment for more complex locomotion testing. NASA has also explored several optimization algorithms from evolutionary algorithms to deep reinforcement learning [43].

At Union College, John Rieffel and his team are developing tensegrity robots that move by vibrating the struts on the tensegrity robots. Between 2013 and 2015, his group developed a wired tensegrity robot that only uses resonant motor frequencies to move forward [17] [18]. The first few gaits of the tensegrity were found by trial-and-error. In Rieffel et al.'s work in 2014, they implemented hill climber to move the robot using hands-off automation of the evolutionary process [18]. In their most recent work, Rieffel et al. propose Bayesian Optimization, a machine learning algorithm, to optimize the gait of their vibrational tensegrity robots [31].

2.2 Bayesian Optimization

Bayesian optimization is, as the name suggests, an optimization algorithm. That is, Bayesian optimization attempts to find the optimal input for some function, which is often the input which maximizes or minimizes the output value of the function. Bayesian optimization is specifically used for determining the optimal input to an unknown function which can be tested but not directly known. In other words, given some unknown function $f(x) : X \to y$ where X is the set of possible inputs and y is a measurable output, Bayesian optimization can find an input x which produces a close-to-optimal output y.

Bayesian optimization utilizes this as a means of estimating the shape of the unknown function it is optimizing. Starting with some prior knowledge, either given to the algorithm a priori or discovered through experimentation, Bayesian optimization generates a Gaussian process which fits the prior knowledge. Then Bayesian optimization refines its model by choosing an input based on its probability distribution using an acquisition function, which examines the Gaussian process and selects the most promising candidate based on criteria which vary between acquisition functions. It then gives this input to the function, measures the output, and fits its Gaussian process to this new datum. That is, it experiments based on probability-based hypotheses and refines its model based on the new empirical evidence it generates. Note that it is assumed

procedure MAP-ELITES ALGORITHM (SIMPLE, DEFAULT VERSION)
    $(\mathcal{P} \leftarrow \emptyset, \mathcal{X} \leftarrow \emptyset)$ ▷ *Create an empty, N-dimensional map of elites: {solutions $\mathcal{X}$ and their performances $\mathcal{P}$}*
    for iter $= 1 \rightarrow I$ **do** ▷ *Repeat for I iterations.*
        if iter $< G$ **then** ▷ *Initialize by generating G random solutions*
            $\mathbf{x}' \leftarrow$ random_solution()
        else ▷ *All subsequent solutions are generated from elites in the map*
            $\mathbf{x} \leftarrow$ random_selection($\mathcal{X}$) ▷ *Randomly select an elite x from the map $\mathcal{X}$*
            $\mathbf{x}' \leftarrow$ random_variation($\mathbf{x}$) ▷ *Create x', a randomly modified copy of x (via mutation and/or crossover)*
        $\mathbf{b}' \leftarrow$feature_descriptor($\mathbf{x}'$) ▷ *Simulate the candidate solution x' and record its feature descriptor $\mathbf{b}'$*
        $p' \leftarrow$performance($\mathbf{x}'$) ▷ *Record the performance p' of x'*
        if $\mathcal{P}(\mathbf{b}') = \emptyset$ or $\mathcal{P}(\mathbf{b}') < p'$ **then** ▷ *If the appropriate cell is empty or its occupants's performance is $\leq p'$, then*
            $\mathcal{P}(\mathbf{b}') \leftarrow p'$ ▷ *store the performance of x' in the map of elites according to its feature descriptor $\mathbf{b}'$*
            $\mathcal{X}(\mathbf{b}') \leftarrow \mathbf{x}'$ ▷ *store the solution x' in the map of elites according to its feature descriptor $\mathbf{b}'$*
    return feature-performance map ($\mathcal{P}$ and $\mathcal{X}$)

Figure 1: A pseudocode description of the MAP-Elites algorithm [24]

that the unknown function is smooth, so that knowledge of the results of some inputs can be used to make judgments about the probability distributions of other inputs. As the algorithm generates more data points and refines its model, it has a better idea of which inputs are likely to be fruitful, and can thus hone in on an optimal solution.

2.3 Map-Elites Algorithm

Multi-dimensional Archive of Phenotypic Elites (MAP-Elites) is a search algorithm designed by Mouret et al. [24]. It creates a map of high-performing solutions at each point in a search space defined by dimensions of variation that a user gets to choose. MAP-Elites algorithm illuminates search spaces, allowing researchers to understand how interesting attributes combine to affect performance. For example, a drug company may wish to understand how performance changes as the size of molecules and their production cost vary. MAP-Elites produces a large diversity of high-performing, yet qualitatively different solutions, which can be more helpful than a single, high-performing solution. Interestingly, because MAP-Elites explores more of the search space, it also tends to find a better overall solution than other state-of-the-art search algorithms [24]. Figure 1 describes the basic design of the MAP-Elites algorithm. It first starts with random selection which randomly generate solutions in the solution space, then after a certain number of iterations, it starts to exploit the random variation of the random solutions found in the random selection phase.

2.4 Research Question

As mentioned in the introduction section, the main goal of this research is to enable Union College's new tensegrity robot (the Wireless Vibrationally Actuated Limbless Tensegrity Robot - VVValtr) to move in diverse ways.

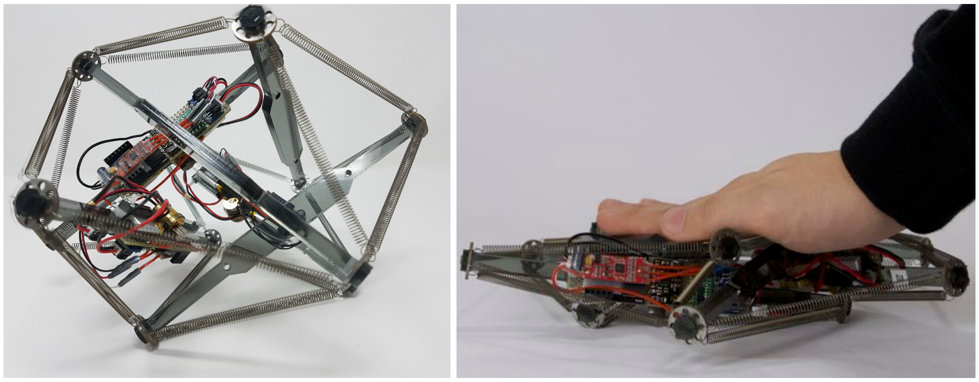

Figure 2: A completely untethered soft tensegrity robot (VVValtr). The robot is resilient enough to be easily compressed by hand.

Figure 2 shows Union College's six-bar tensegrity robot VVValtr. This canonical icosahedron shape consists of six struts (3 active and 3 passive) connected by 24 springs. By using a bluetooth RFduino installed on each active strut, we are able to vibrate the on-board motors according to a resonant frequency command. Hence, for this robot, a movement command is a combination of three numbers, one frequency for each motor. We are able to control the time of vibration on the computer by sending a stop command. Since the frequency is not changed during one control period, the control type of the robot is open-loop. The task of achieving effective and interesting locomotion is to find the frequency combinations that achieve the best rotation and translation. With this task in hand, the question this research seeks to address is **to what extent can we efficiently explore the behavior space of our tensegrity robot to achieve more efficient and more diverse locomotion?**

3 Generate gaits with longest translation using Bayesian Optimization

My goal is to learn how to exploit the mechanics of tensegrity robots to make them move in multiple directions, but we do not know what is the best optimization algorithm in order to generate an optimal gait for the newly designed wireless tensegrity robot. With this research, I am trying to learn and implement different control and optimization methods to help our wireless tensegrity robot (VVValtr) gain effective locomotion. Specifically, I will explore how Bayesian Optimization and MAP-Elites could affect the locomotion of VVValtr. I will analyze the performance by considering the time and energy each solution take relative to solutions that achieve similar locomotion results to ultimately determine which movement strategy is the best based on different features.

As in a previous student's thesis [2] and also in prior work of ours [31] Bayesian Optimization was employed to optimize locomotion on a tethered tensegrity robot. Here we extend those results on a fully autonomous and tether-free robot, using the off-the-shelf BO package pyGPGO [15]. The goal of this particular experiment is to optimize the choice of frequency combinations to find gaits that enable the robot to translate in the longest distance.

3.1 Tensegrity Robot Tracking

We control VVValtr by sending frequency speeds to the motors mounted on the rigid struts via the on-board RFduino. Since we only care the scalar measurement of the distance traveled of the tensegrity, we measure the distance in each iteration of experiment by subtracting the starting and ending locations of the robot obtained from a camera mounted above the testing table. However, this method cannot measure the rotation of each locomotion. In my research, we first use the OpenCV library and build a tracking system based on finding the contour of the robot and then find the central point. By doing this, we are able to know the position of the robot on the testing table at any time.

3.2 Comparing Random Search and Bayesian Optimization

Since my research is trying to explore different ways to achieve diverse open-loop control of our tensegrity robot, I implemented the original hill-climber search algorithm which uses random values to find gaits for VVValtr. By doing this, we will be able to get basic benchmarks of translations.

Bayesian Optimization (BO), as introduced, fits a probabilistic model (in this case a Gaussian process [28]) that maps motor speeds to locomotion distance.

We used the suggested setup by Boggs [2]. In his research, he showed promising result on his experiment setup. However, with limited time and resources, he didn't fully test his experiment setup with sufficient iterations of the experiment.

In his setup, the testing arena consists of a level surface on which the tensegrity could move, and a USB camera located above it in order to track the movement of the tensegrity using OpenCV. After each iteration, trial motor frequencies chosen by the optimizer running on a control computer were sent to the robot, which would then be allowed to run for 30 seconds, after which our motion capture system reported the distance the tensegrity traveled back to the optimizer computer. This information in turn was used by the optimizer to produce new sample frequencies.

3.3 Experiment Details & Results

We ran experiments with the aim of maximizing linear displacement of the robot over a 30 second trial period. Only the radial distance was used for evaluation, regardless of the actual path traveled.

Search Protocol	# trials/run	# runs
BO, β=0.5	60	10
BO, β=1.0	60	10
BO, β=1.5	60	10
BO, β=2.0	60	10
BO, β=2.5	60	10

Table 1: Details of the β searching trials ran in order to determine the most effective β value [20]

Search Protocol	# trials/run	# runs
BO with priors	60	30
Random Search	60	30
BO without priors	60	30

Table 2: Details of the searching trials ran in order to determine and compare the effectiveness of search policies [20]

.

As shown in Table 1 and 2, I worked collaboratively with Jonathan Kimber in the 2018 Fall [20]. We first conducted some initial beta value searching. In this beta searching period, we had 60 consecutive learning trials for a single run, at which point we felt we had suitably optimized the gaits for our robot. Each run took approximately 1 hour. We conducted 10 independent runs for each beta value, and found that $\beta = 2.0$ (shown in Figure 3) provided the best balance between exploration and exploitation in order to efficiently find effective gaits.

We then run experiments to compare different search policies: Bayesian Optimization (with priors biased in favor of high motor speeds), cold-start Bayesian Optimization (without priors), and completely random search. For each of these treatments, we ran each optimization for 60 evaluations (each 30 seconds) per trial. We then repeated each experiment across 30 trials in order to gather a representative sample.

The results, as shown in Figure 4, are somewhat equivocal in regards to Bayesian Optimization, although with several points. Figure 5 shows the progress over the 60 trials of each search policy. In each case, search policies were able to find motor frequency sets that produced significant linear motion (as measured by displacement). Among the three methods, BO produced some of the quantitatively fastest gaits (3+ body lengths), although a statistical comparison suggests that it does not significantly outperform the other search methods. This is because the valid frequency range is relatively small for BO process to explore. Thus, only conducting open-loop control with BO will not significantly improve the searching [20].

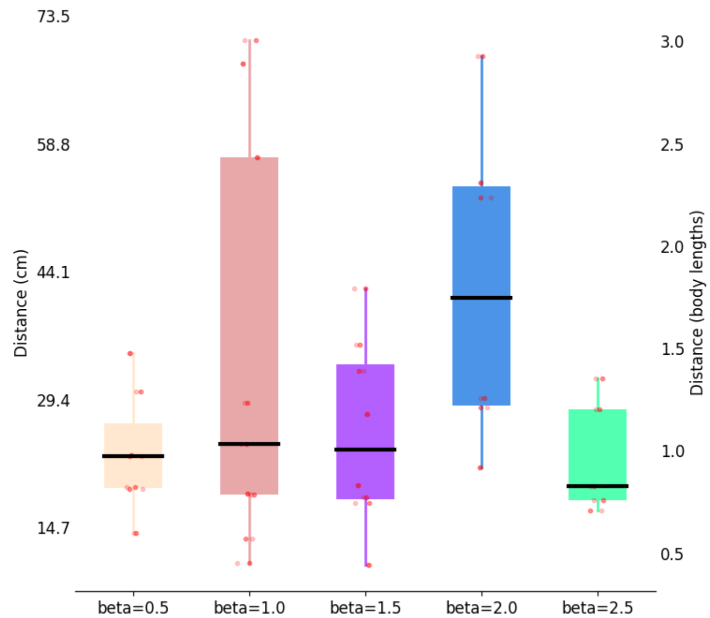

Figure 3: Shows β=2.0 is the most effective for movement.

4 Exploring the Behavior Repertoire using MAP-Elites algorithm

4.1 Qualisys Tracking System

Thanks to the new NSF grant in the CROCHET lab, during the 2018 Winter break, the lab was installed with a new 3D motion capture system called Qualisys. It uses the advanced tracking cameras mounted on the roof of the crochet lab to provide motion data with sub-millimeter accuracy. It has a built-in Qualisys Software Development Kit (SDK) that could allow us to live stream position data of rigid body that we defined with Infrared Reflected (IR) markers.

After connecting with the Qualisys tracking system, we first attached IR markers to various struts of our tensegrity robot. Then, by defining the rigid body of these struts, we are able to get the position data (center of mass) and rotation matrices of each strut relative to the global frame. Figure 5 is a how VVValtr is presented in the tracking system.

We then use these live streamed data to calculate the center of mass of the entire robot and the Euler angle with respect to the z-axis of the global frame to enable tracking of the Valtr with both translation and rotation information.

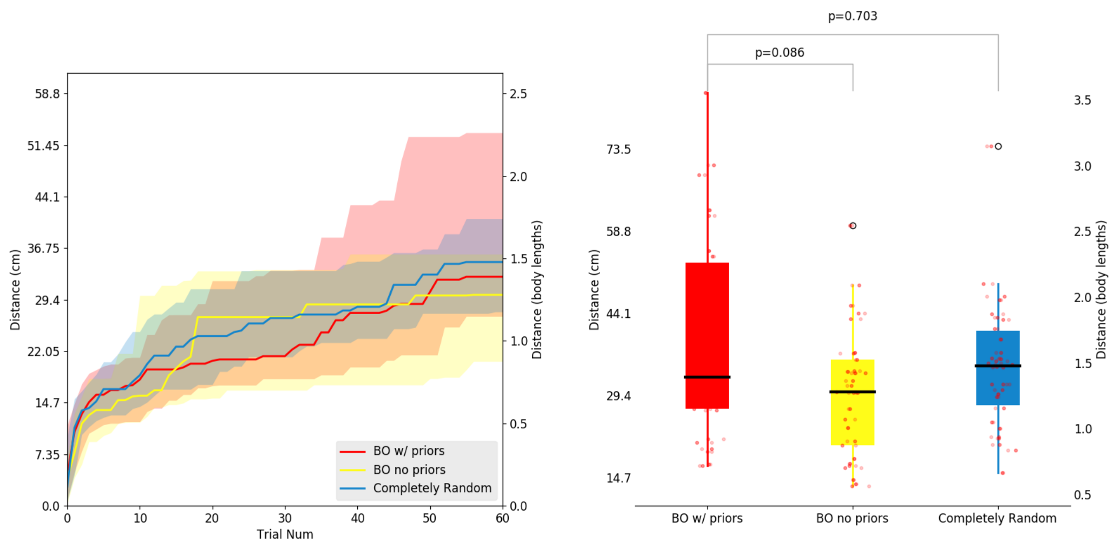

Figure 4: The line plot on the left shows the value of the best-so-far evaluation of each search policy, across 60 evaluation steps (30 independent trials apiece), with shaded quartile intervals. The box plot on the right shows the final distribution of the best distance after 60 evaluations [20]

.

4.2 MAP-Elites experiment

The main reason that we are trying to implement this MAP-Elites algorithm on our tensegrity is that researchers have implemented it on the legged robots and successfully found effective locomotion [9]. We believe MAP-Elites can also help VVValtr to explore its behavior space.

4.2.1 Behavior of Valtr in the Behavior Space

Figure 6 is a simple example of how to implement a single iteration of MAP-Elites on VVValtr. By giving a solution which is a combination of three motor frequencies, we then send this set of frequencies to the VVValtr via the on-board RFduino. With our updated tracking method, we will be able to effectively identify the change in translation and rotation of VVValtr.

MAP-Elites algorithm is a kind of Quality Diversity (QD) algorithm which finds a mapping between the parameter space (frequency commands) and behavior space (robot positions and orientations) based on the quality metric (time efficiency or energy efficiency).

We first define the behavior space of VVValtr like Figure 7 as the translation coordinates on the $x - y$ plane and then the rotation with respect to its own orientation. Each point on the behavior space describes, under a certain frequency combinations, after certain time t, the robot moves to a point on $x - y$ plane with its original placement as origin of the behavior space. As the behaviors in the behavior space are described

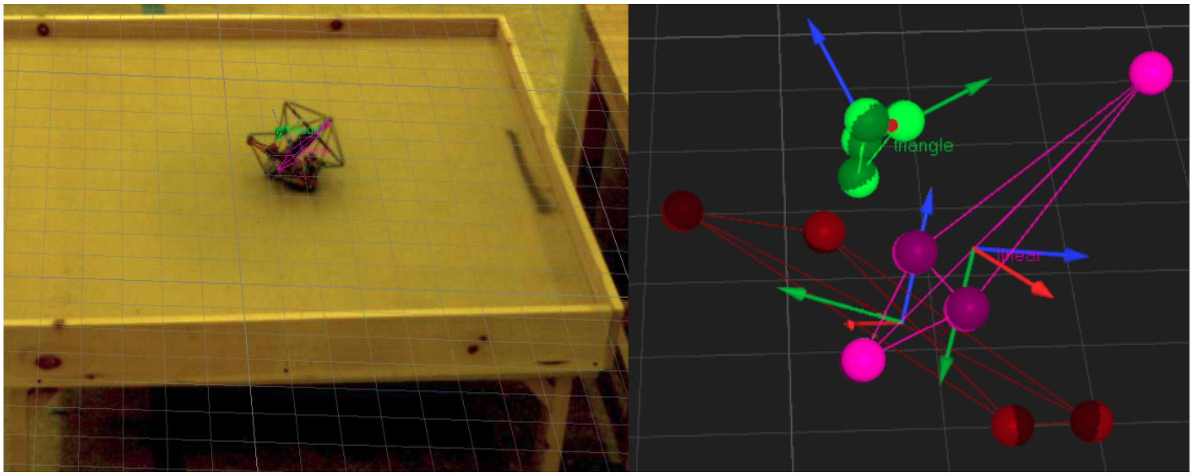

Figure 5: A figure shows VVValtr in the color camera and the Qualisys SDK with three struts defined as rigid bodies and their center of mass shown as small x-y-z axis.

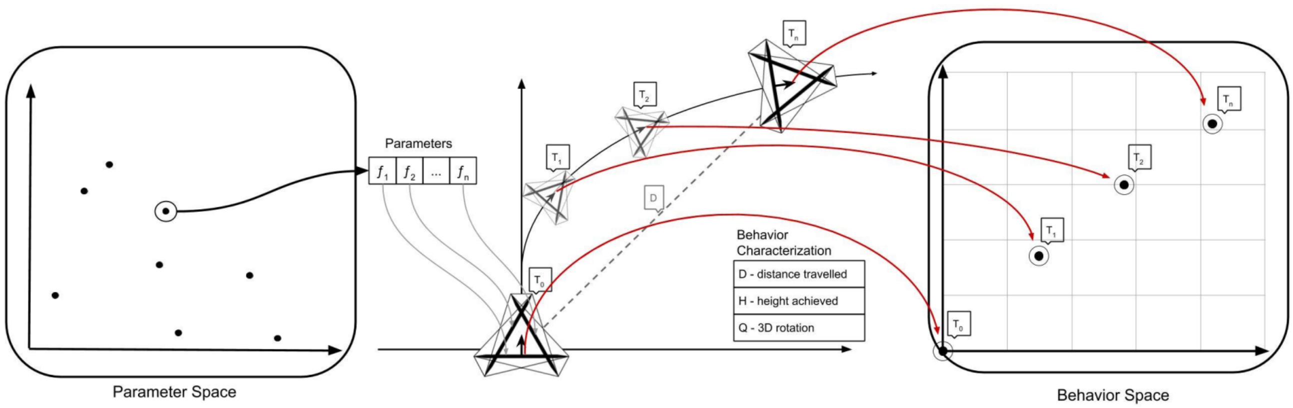

Figure 6: A figure by Rieffel [30] that illustrates how the quality diversity algorithm works on VVValtr by tracking translation and rotation behaviors over different time periods.

in a continuous manner, in order to describe or categorize each behavior, we discretize the behavior space into small grids. In other words, we define rotating 45 degrees as same as rotating 45.5 degrees. We define moving to position $(10, 10)$ as same as moving to position $(10.5, 10.5)$.

4.2.2 Behavior Repertoire of Valtr

Since each solution (frequency combination) would generate a behavior in behavior space at certain time t, then different solutions would generate various behaviors at various times. The **Behavior Repertoire** of our robot would be the diverse behaviors that are generated by different solutions at different times t.

The MAP-Elites algorithm, in our problem context, maps a set of frequency combinations to a variety of locomotion abilities of our tensegrity robot. In the context of the introduction of MAP-Elites in the back-

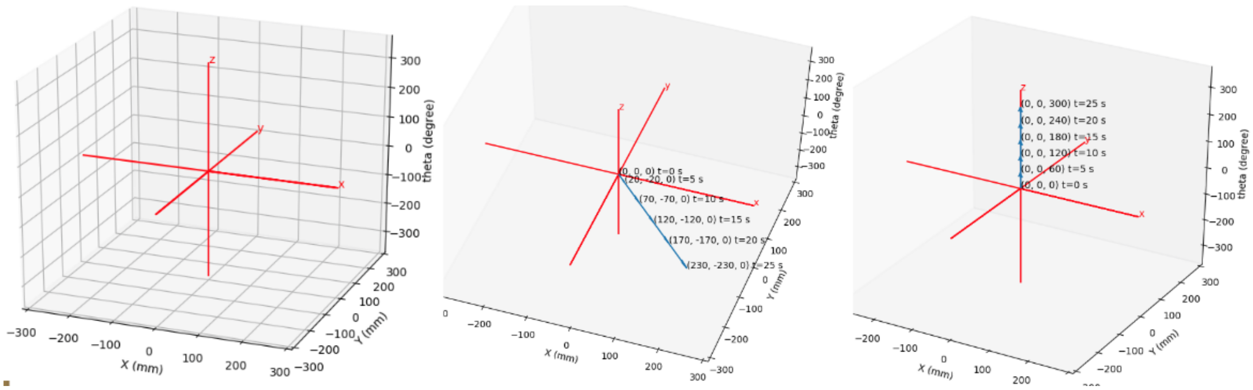

Figure 7: Leftmost picture is a 3D behavior space that has translation in the x-y plane as the 2D component and the rotation of robot with respect to it starting orientation as the third dimension. The picture in the middle is a behavior of robot with only translation with no rotations. The picture on the right is a behavior of robot that only rotates but does not translate.

ground section, a solution to the problem is a combination of motor frequencies. The features that we are interested in are rotations and translations. The performance measurement of the solution is how much time and energy it takes for the solution to achieve such a translation or rotation. In our case, shorter time means better performance; when the time ties, the solution using the least energy means better performance.

In other words, as we are able to use different solutions to filled up VVValtr's behavior space to yield a **Behavior Repertoire**, however, there will be lots of overlaps of points in our **Behavior Repertoire** since lots of frequency solutions will yield similar behaviors. Here, MAP-Elites algorithms could help us to find the best solution (frequency combination) among these overlaps according to our metric which in this case, the time and energy for VVValtr to achieve this behavior under this solution.

In short, imagine solution $a = (f_1, f_2, f_3)$ and solution $b = (f'_1, f'_2, f'_3)$ both move Valtr to point (x, y, z) in behavior space. If a takes less time to do this than b, then a is better; the **Behavior Repertoire** will have a as the solution that could achieve this behavior. However, if a takes the same amount time as b, but b takes less energy (the square sum of the three frequencies, e.g. $(f_1)^2 + (f_2)^2 + (f_3)^2 > (f'_1)^2 + (f'_2)^2 + (f'_3)^2)$, then the **Behavior Repertoire** will keep b as the solution to this behavior. This is also how MAP-Elties updates the "better" solutions on the behavior space with user defined performance metric.

With the updated policy and the basic setup of every element described in the MAP-Elites algorithm, we will be able to fill the behavior space by conducting the experiments. Then based on the plot we get after multiple iterations of search and update, we will have solutions (frequency combination) that moves tensegrity in desired motion in a short amount of time.

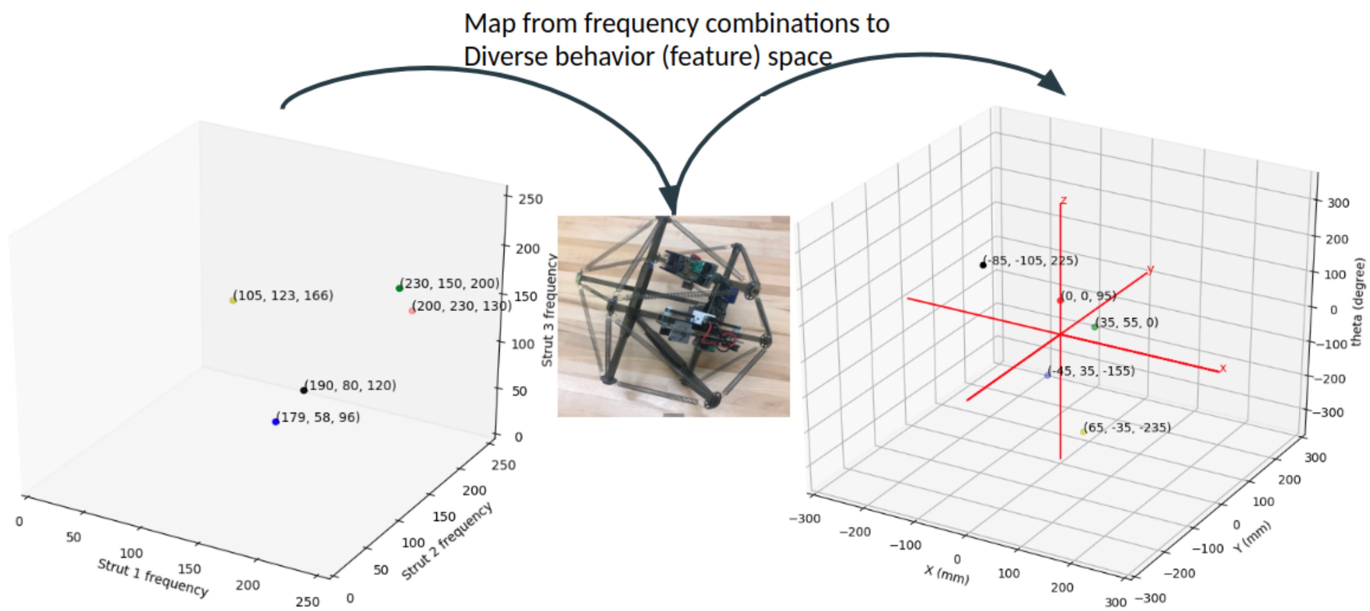

Figure 8: This is a simple illustration on how MAP-Elites solving the problem in our context. It maps a solution which is a combination of three motor frequencies to the behavior space which describes the locomotion of the robot.

4.2.3 Experiment details

We did 360 iterations of MAP-Elites algorithm. Among these 360 iterations, 200 iterations are random sampling of random frequency combination solutions. The other 160 iterations are random variation of 200 tested solutions.

In each iterations, we allow VVValtr to run 30s and track the 7 points with 5 seconds interval. Thus, we know the position of VVValtr at $t = 0, 5, 10, 15, 20, 25, 30$. We then describe the $t = 0$, as $(0, 0, 0)$ in the behavior space and describe the behavior of $t > 0$, as the positions and Euler angles subtract from the position and Euler angle from $t = 0$.

After 360 iterations, we generated $360 * 7 = 2520$ points in the behavior space. Then, we categorized these points into grids and chose the solution with the least time and energy consumption to form our **Behavior Repertoire**.

4.3 Results & Data Visualization

After these 360 iterations with 200 random sampling and 160 random variations, we observed lots of interesting behaviors with in these 30s trials.

Figure 9 to Figure 12 shows the various motions from these 360 iterations of **Behavior Repertoire** exploration.

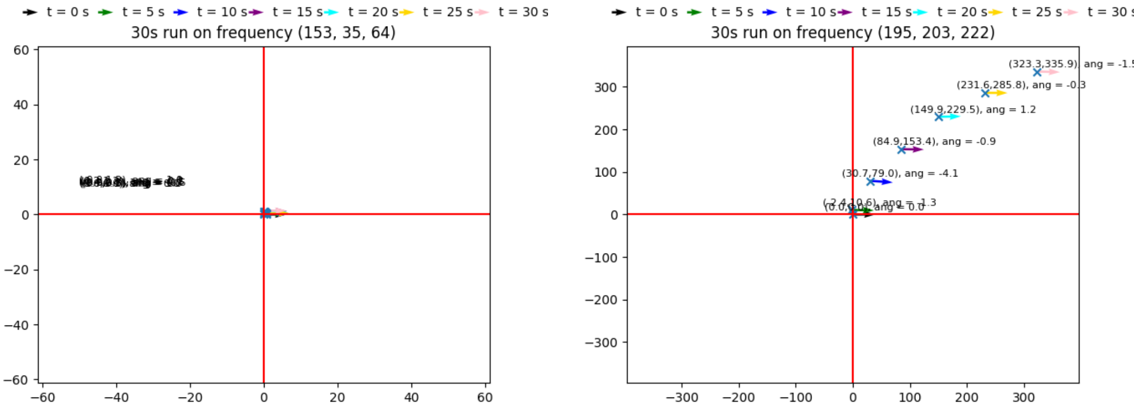

Figure 9: A 2D illustration of not moving solution and linear movement solution.

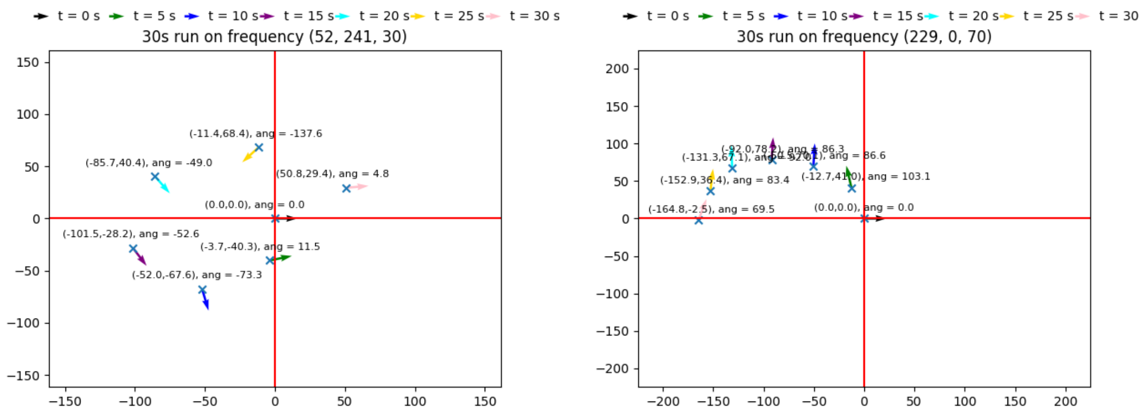

Figure 10: A 2D illustration of circular movement solution and arc movement solution.

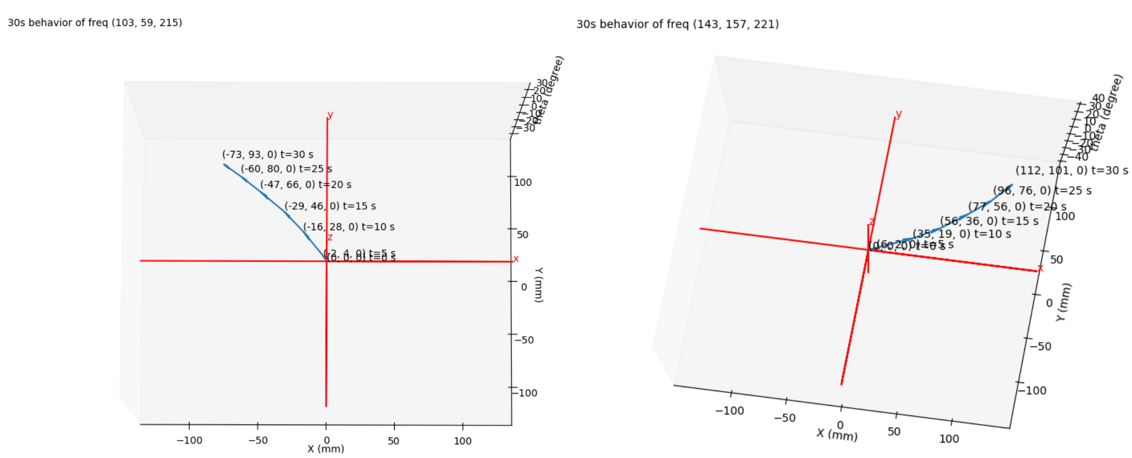

Figure 11: A 3D illustration of linear movement solution.

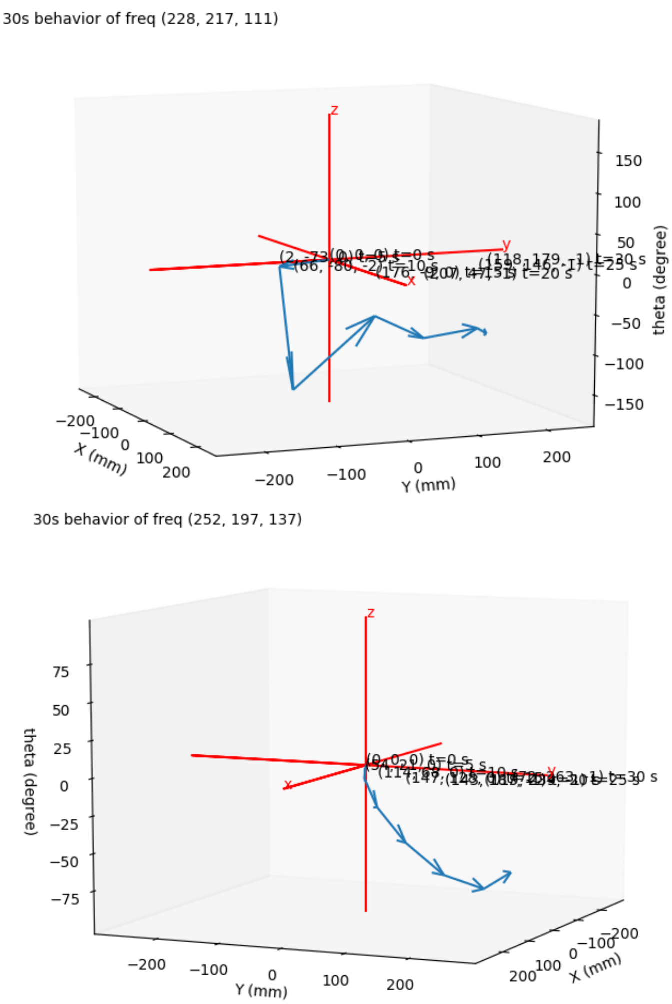

Figure 12: A 3D illustration of rotation movement solution.

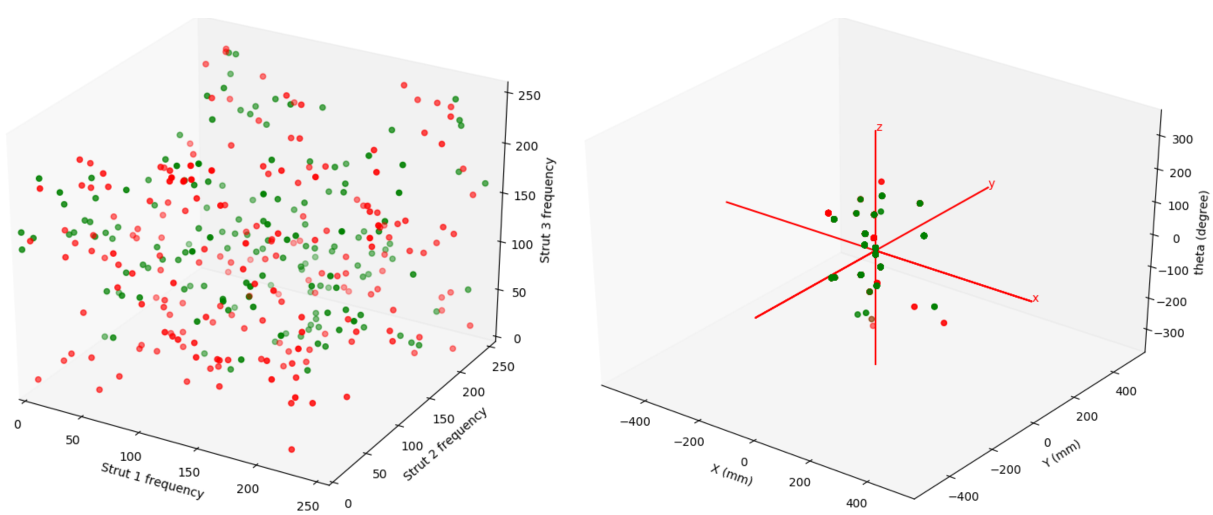

Figure 13: Final Solutions space and its **Behavior Repertoire** after 360 iterations.

We discretized the behavior space into 53428 total grids (38 * 38 * 39). Among the $2520(360 * 7)$ points generated by this experiment, the 200 random sampling iterations with 1400 points illuminated 618 grids, generated similar behavior 582 times. The 160 random variation iterations with 1120 points illuminated 375 grids, generated similar behavior 745 times.

Thus, the total number of points in the **Behavior Repertoire** is 993. Thus, after 360 iterations of this MAP-Elites experiment, we illuminated 993 out of 53428 total grids (1.8 %).

Since the total number of data taken by the time of generating these figures were only 360, it could not show any effectiveness of the MAP-Elites methods on finding more diverse behaviors of tensegrity or showing a comprehensive version of the **Behavior Repertoire**.

However, as we could tell from these figures, especially from the single iteration figures, our MAP-Elites experiment caught lots of interesting and diverse behaviors of our tensegrity robot. We also see promising results on how much the behavior space is being filled up after only 360 iterations. It is worth noticing that the behavior space grid that we define is a grid with high precision, that is why the **Behavior Repertoire** that we generated seems to be cluttered in the center of the behavior space.

In conclusion, with the newly installed Qualisys Tracking System in our lab environment, we developed the way to track our wireless tensegrity robot Valtr in the global frame of our lab. Then based on the suggestions from the previous Bayesian Optimization gait generation work [20], I developed and implemented the MAP-Elites [24] experiment for our tensegrity robot incorporating the new Qualisys tracking system. Then with few initial search in our behavior space, we found lots of interesting and diverse behaviors of our tensegrity including, rotation, linear, arc, and circular movements. We also built-up a preliminary version of **Behavior Repertoire** of our own wireless tensegrity robot for future control and reference.

5 Contributions

I joined the Union College tensegrity group during my sophomore year and started to assist different parts on the tensegrity research. With the guidance from Professor John Rieffel and the help of other team members. The contributions of this research project were significant to improve our tensegrity research. I first used Hill-Climber algorithm to make our tensegrity move and recorded a few frequency combinations that translate or rotate VVValtr as benchmarks. Then I used the Bayesian Optimization algorithm to find different frequency solutions to move the tensegrity and set the goal of the optimization to move the robot the longest distance. Recorded the frequency combinations that translate VVValtr the greatest distance. At the end of the first phase of this research, I contributed to a peer reviewed paper that will be published in an International Conference [20]. During Winter 2019, I created tracking methods for VVValtr with IR markers using the Qualisys Tracking System. Then I Implemented the MAP-Elites algorithm on VVValtr and tried to explore the behavior space which combines rotations and translations of VVValtr. Ran more trials to find the optimal solution for each point in the behavior space. I also developed the data visualization tool for the experimental results from MAP-Elites experiment.

6 Concerns and Future Work

Even though the testing setup and environment were significantly improved after the lab was equipped with the Qualisys Tracking System, we are still seeing some unstable design in the testing system.

First, from the tracking of the tensegrity robot, since the experiment of MAP-Elites [24] in our context requires us to track the robot positions in the global frame during the vibration movement of the robot, the tracking system some times could not catch the rigid body defined by the struts when they vibrate violently or if the system receives data with high uncertainties.

Secondly, the software on the on-board RFduino hasn't been updated since the work done by Boggs [2]. This code has a lot of system check up processes that were designed for the Bayesian Optimization experiment; it sometimes would cause the re-run of the previous iteration's frequency command. In the MAP-Elites experiment, I caught these errors by if-then statements and pausing the experiment automatically to re-test the desired frequency combination. The on-board Ardurino code should be updated with more flexible design for different types of experiments.

The last concern from my research with VVValtr is the mechanical part. With different energy left in the battery, the motor behavior on each strut seems inconsistent with respect to the same frequency command. This is only a concern that is yielded by human observations. We need to have a more rigorous test on each strut to determine whether the circuitry of the strut needs to be improved or updated again.

All the above control strategies are open-loop control which do not change the action of the robot based on the environment and current state of the robot. However, with the on board accelerometer and encoder, we are able to sense the current state of the robot and enable it to act according to the surrounding environment. Hence, by doing such sensing, we are able to install a controller like CPG [1] or Spiking Neural Networks [33] on the tensegrity robot itself to achieve closed-loop control.

There are several refinements we look forward to implementing - specifically in regards to the type of control algorithm and corresponding learning algorithms. While the open-loop controller used in the results above was surprisingly effective, the dynamical complexity of the tensegrity robot, coupled with the resonance-based locomotion scheme, imposes considerable variability on the performance of the robot. Small differences in the composition of the surface, and of the initial phase of the motors, can have significant effects upon behavior. As a consequence, we will next be exploring closed-loop control strategies such as neural networks and CPGs to control the robot, with feedback coming in from both a high speed 3D motion capture system as well as on-board sensors such as shaft encoders and accelerometers. These, coupled either with conventional reinforcement learning or with data-efficient black-box policy search algorithms [7], should allow us to produce highly robust and dynamical gaits. We are trying to utilize a new closed-loop implementation using a novel reinforcement learning algorithm called BLACK-DROPS [7].

7 Conclusions

Although not conclusive, the research described here has paved the way for increasingly interesting and varied investigations into the behavior diversity of tensegrity robots. First, we have developed a new, modular tensegrity robot equipped with data collection and wireless communications capabilities, a significant advance from our previous tensegrity robot. We utilized these new capabilities and fully tested the work suggested previously by research on tensegrity gait generation undertaken by Rieffel et el. [31] and Boggs [2] that Bayesian optimization is a promising means of generating effective gaits on tensegrity robots and generated publishable results [20]. The work [20] that we published suggested the necessity of exploring more diverse behavior of tensegrity. Following this suggestion, we have a new advanced tracking system to improve our testing environment. In particular, we have demonstrated a thorough experiment design and implementation with some preliminary data that shows some beautiful and effective behavior of our new wireless tensegrity robot, and specifically that some of the solutions could be used in the future to generate a desired control sequence like moving steadily forward. Equally important, through this research we have discovered a number of flaws and failures in our new setup which must be addressed, and in doing so paved the way for more effective research in the future.

References

[1] Thomas Bliss, Jeffrey Werly, Tetsuya Iwasaki, and Hilary Bart-Smith. Experimental validation of robust resonance entrainment for cpg-controlled tensegrity structures. *IEEE Transactions On Control Systems Technology*, 21(3):666–678, 2013.

[2] J. Boggs. Optimizing tensegrity gaits using bayesian optimization.

[3] Valter Böhm and Klaus Zimmermann. Vibration-driven mobile robots based on single actuated tensegrity structures. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 5475–5480. IEEE, 2013.

[4] Jonathan Bruce, Ken Caluwaerts, Atil Iscen, Andrew P Sabelhaus, and Vytas SunSpiral. Design and evolution of a modular tensegrity robot platform. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 3483–3489. IEEE, 2014.

[5] Ken Caluwaerts, Jérémie Despraz, Atıl Işçen, Andrew P Sabelhaus, Jonathan Bruce, Benjamin Schrauwen, and Vytas SunSpiral. Design and control of compliant tensegrity robots through simulation and hardware validation. *Journal of the royal society interface*, 11(98):20140520, 2014.

[6] Wai Leung Chan, Diego Arbelaez, Frederic Bossens, and Robert E Skelton. Active vibration control of a three-stage tensegrity structure. In *Smart Structures and Materials 2004: Damping and Isolation*, volume 5386, pages 340–347. International Society for Optics and Photonics, 2004.

[7] Konstantinos Chatzilygeroudis, Roberto Rama, Rituraj Kaushik, Dorian Goepp, Vassilis Vassiliades, and Jean-Baptiste Mouret. Black-Box Data-efficient Policy Search for Robotics. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2017.

[8] Robert Connelly and Allen Back. Mathematics and tensegrity: Group and representation theory make it possible to form a complete catalogue of" strut-cable" constructions with prescribed symmetries. *American Scientist*, 86(2):142–151, 1998.

[9] Miguel Duarte, Jorge Gomes, Sancho Moura Oliveira, and Anders Lyhne Christensen. Evolution of repertoire-based control for robots with complex locomotor systems. *IEEE Transactions on Evolutionary Computation*, 2017.

[10] Etienne Fest, Kristina Shea, and Ian FC Smith. Active tensegrity structure. *Journal of Structural Engineering*, 130(10):1454–1465, 2004.

[11] Richard Buckminster Fuller. Non-symmetrical tension-integrity structures, February 18 1975. US Patent 3,866,366.

[12] Ariel Hanaor. Aspects of design of double-layer tensegrity domes. *International Journal of Space Structures*, 7(2):101–113, 1992.

[13] Donald E Ingber. Cellular tensegrity: defining new rules of biological design that govern the cytoskeleton. *Journal of cell science*, 104(3):613–627, 1993.

[14] Atil Iscen, Adrian Agogino, Vytas SunSpiral, and Kagan Tumer. Learning to control complex tensegrity robots. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 1193–1194. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[15] José Jiménez and Josep Ginebra. pygpgo: Bayesian optimization for python. *The Journal of Open Source Software*, 2:431, 2017.

[16] Sergi Hernandez Juan and Josep M Mirats Tur. Tensegrity frameworks: static analysis review. *Mechanism and Machine Theory*, 43(7):859–881, 2008.

[17] Mark Khazanov, Ben Humphreys, William D Keat, and John Rieffel. Exploiting dynamical complexity in a physical tensegrity robot to achieve locomotion. In *ECAL*, pages 965–972. Citeseer, 2013.

[18] Mark Khazanov, Jules Jocque, and John Rieffel. Evolution of locomotion on a physical tensegrity robot. In *ALIFE*, volume 14, pages 232–238, 2014.

[19] Kyunam Kim, Deaho Moon, Jae Young Bin, and Alice M Agogino. Design of a spherical tensegrity robot for dynamic locomotion. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 450–455. IEEE, 2017.

[20] Petridou Sipple Barhydt Booigs Dosiek Kimber, Ji and Rieffel. Low-cost wireless modular soft tensegrity robots. *Robosoft 2019 - IEEE International Conference on Soft Robotics*, 2019.

[21] Yuusuke Koizumi, Mizuho Shibata, and Shinichi Hirai. Rolling tensegrity driven by pneumatic soft actuators. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1988–1993. IEEE, 2012.

[22] Stephen M Levin. The tensegrity-truss as a model for spine mechanics: biotensegrity. *Journal of mechanics in medicine and biology*, 2(03n04):375–388, 2002.

[23] Takeshi Matsuda and Satoshi Murata. Stiffness distribution control-locomotion of closed link robot with mechanical softness. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 1491–1498. IEEE, 2006.

[24] Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*, 2015.

[25] TW Myers. Anatomy trains: Myofascial meridians for manual and movement therapists, churchill livingstone elsevier, 2009.

[26] Chandana Paul, Francisco J Valero-Cuevas, and Hod Lipson. Design and control of tensegrity robots for locomotion. *IEEE Transactions on Robotics*, 22(5):944–957, 2006.

[27] Anthony Pugh. *An introduction to tensegrity*. Univ of California Press, 1976.

[28] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.

[29] Colin M Rennie. *Designing and learning CPG gaits for spherical tensegrity robots using Bayesian optimization*. PhD thesis, Rutgers University-School of Graduate Studies, 2018.

[30] John Rieffel. Adaptable behavioral repertoiresfor resilient soft tensegrity robots.

[31] John Rieffel and Jean-Baptiste Mouret. Adaptive and resilient soft tensegrity robots. *Soft robotics*, 2018.

[32] John Rieffel, Francisco Valero-Cuevas, and Hod Lipson. Automated discovery and optimization of large irregular tensegrity structures. *Comput. Struct.*, 87(5-6):368–379, March 2009.

[33] John A Rieffel, Francisco J Valero-Cuevas, and Hod Lipson. Morphological communication: exploiting coupled dynamics in a complex mechanical structure to achieve locomotion. *Journal of the royal society interface*, 7(45):613–621, 2010.

[34] Andrew P Sabelhaus, Jonathan Bruce, Ken Caluwaerts, Pavlo Manovi, Roya Fallah Firoozi, Sarah Dobi, Alice M Agogino, and Vytas SunSpiral. System design and locomotion of superball, an untethered tensegrity robot. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 2867–2873. IEEE, 2015.

[35] Mizuho Shibata, Fumio Saijyo, and Shinichi Hirai. Crawling by body deformation of tensegrity structure robots. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4375–4380. IEEE, 2009.

[36] Robert E Skelton and Mauricio C de Oliveira. *Tensegrity systems*, volume 1. Springer, 2009.

[37] Robert T Skelton and Cornel Sultan. Controllable tensegrity: a new class of smart structures. In *Smart Structures and Materials 1997: Mathematics and Control in Smart Structures*, volume 3039, pages 166–178. International Society for Optics and Photonics, 1997.

[38] Cornel Sultan, Martin Corless, and Robert E Skelton. Tensegrity flight simulator. *Journal of Guidance, Control, and Dynamics*, 23(6):1055–1064, 2000.

[39] Vytas SunSpiral, George Gorospe, Jonathan Bruce, Atil Iscen, George Korbel, Sophie Milam, Adrian Agogino, and David Atkinson. Tensegrity based probes for planetary exploration: Entry, descent and landing (edl) and surface mobility analysis. *International Journal of Planetary Probes*, 7, 2013.

[40] AG Tibert and Sergio Pellegrino. Review of form-finding methods for tensegrity structures. *International Journal of Space Structures*, 26(3):241–255, 2011.

[41] Bin-Bing Wang. Cable-strut systems: part itensegrity. *Journal of Constructional Steel Research*, 45(3):281–289, 1998.

[42] Davide Zappetti, Stefano Mintchev, Jun Shintake, and Dario Floreano. Bio-inspired tensegrity soft modular robots. In *Conference on Biomimetic and Biohybrid Systems*, pages 497–508. Springer, 2017.

[43] Marvin Zhang, Xinyang Geng, Jonathan Bruce, Ken Caluwaerts, Massimo Vespignani, Vytas SunSpiral, Pieter Abbeel, and Sergey Levine. Deep reinforcement learning for tensegrity robot locomotion. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 634–641. IEEE, 2017.